야후 파이낸스에서 쿨섹펀하게 크롤링하자

구조 가 너무 좆같아서 정리를 하고 가야겠다.

목표는

statistics에 있는 지표를 크롤링할 것

tbody는 총 10개

|  |
| --- |
| valuation |
| stock price history |
| share statistics |
| fiscal year |
| profitable |
| management effectiveness |
| I/s |
| div &split |
| balance sheet |
| cf |

가있다.

다하면 좋아~

stastic 폴더에 몰아 넣자

크롤링 순서

#라이브러리 불러오기

from bs4 import BeautifulSoup

import requests

import pandas as pd

import os

#ticker호출

os.chdir('C:\\Users\\55754\\Documents\\야후파이낸스 재무재표')

ticker=pd.read\_csv('data/ticker/ticker.csv', index\_col=0, header=0)

symbol=[ticker['symbol']]

#for loop해야함

그안에서도 for 해야함

##tbody구조 파해치기

<tbody>테이블

<tr>행

<td> 열</td>

<td> 열</td>

</tr>

…

<tr> 행

<td> 열</td>

<td> 열</td>

</tr>

</tbody>

이런 tbody가 10개 있음

그러니깐

for a in table:

for b in tr:

이지랄 해야한다.

=>

for a in table(=soup.find\_all(‘tbody”)

l=[]

=>첫번째 루프에서는 valuation 지표 크롤링

여기서 for문 한번더 걸어줌

어차피 한가지에 담을 꺼니깐

리스트를 만들고 나서

for 테이블문

for 행문

이렇게담는다.

배워야 할 것 존나 조그만한 반복문

내포리트

난관

4.27B

이거 어떻게 할까

로직

4.27 + B 로 분리하고 이것을 왼쪽을 숫자 오른쪽을 000000000으로 치환하여 바꿈

코드 분석

##########################################################################

#for i in symbol:

for i in symbol:

#포메팅으로 url 하기

url='https://finance.yahoo.com/quote/%s/key-statistics?p=%s' %(i,i)

#html문서 get

r=requests.get(url)

#파서+ 뷰티풀수프

soup=BeautifulSoup(r.text,"lxml")

#리스트 만들기

l=[]

#테이블당 크롤링

for a in soup.find\_all('tbody'):

#행당 크롤링

for b in a.find\_all('tr'):

#열로 나누기

td=b.find\_all('td')

content=[td[0].text, td[1].text]

#리스트에 추가하기

l.append(content)

#데이터 프레임 만들기

df=pd.DataFrame(l,columns=["index",'value'])

#데이터 처리

df['value']=df['value'].str.replace("N/A",'NA')

if True in df['value'].str.endswith("M") :

df['value'][df['value'].str.endswith("M")]=(pd.to\_numeric(df['value'][df['value'].str.endswith("M")].str.replace('M',""))\*1000000).astype("str")

if True in df['value'].str.endswith("B") :

df['value'][df['value'].str.endswith("B")]=(pd.to\_numeric(df['value'][df['value'].str.endswith("B")].str.replace('B',""))\*1000000000).astype("str")

df.to\_csv('data\\statics\\%s.csv' %i)

if True in df['value'].str.endswith("M") :

df['value'][df['value'].str.endswith("M")]=(pd.to\_numeric(df['value'][df['value'].str.endswith("M")].str.replace('M',""))\*1000000).astype("str")

이함수 전격 분석!!! 좌항

df['value'][df['value'].str.endswith("M")]

df [인덱스1] [인덱스2]

[인덱스2]:

df['value'].str.endswith("M")

str.endswith 는 인자를 끝문자에 포함하고 있는 행을 T로 반환

우항

(pd.to\_numeric(df['value'][df['value'].str.endswith("M")].str.replace('M',""))\*1000000).astype("str")

겉에서부터 안으로

(pd.to\_numeric( )).astype(“str”)

정수화 한뒤 다시 문자열로

=>안쪽 함수

(df['value'][df['value'].str.endswith("M")].str.replace('M',""))

df['value'][df['value'].str.endswith("M")].=>data.frame

data.frame.str.replace(“m”,””)

이런 함수가 있다 정도